IN THE CLAIMS

Please amend the claims as follows:

1. (Currently Amended) A computer-implemented method of storing data in a first database, the method comprising:

storing data input in a data entry format via an interface, wherein the data in the data entry format excludes information required by a data entry rule;

transforming the stored data from the data entry format to a first data storage format, wherein the data in the first data storage format includes the information required by the data entry rule;

identifying an error in the data in the first data storage format, the error belonging a particular type of error;

routing the data to a selected one of first and second error correctors based on a type of the error;

receiving corrected data from the selected one of the first and second error correctors; [[~~and~~]]

storing the corrected data in the first database in the first data storage format;

receiving data identifying a user who input the data via the interface;

deriving additional data to be stored in the first data storage format based on the input data and based on an identity of the user; and

storing the additional data in the first database.

2. (Previously Presented) The method of claim 1, wherein transforming is performed by a rules-based procedure.

3. (Previously Presented) The method of claim 1, further comprising providing default data values in the interface.

4-5. (Cancelled)

6. (Currently Amended) The method of claim [[4]] 1, further comprising defining dynamically the data entry format based on an identity of the user.

7. (Previously Presented) The method of claim 6, further comprising providing default data values to the interface.

8. (Previously Presented) The method of claim 1, further comprising:

transforming the data from the data entry format to a second data storage format; and

storing the data in a second database in the second data storage format.

9. (Currently Amended) A computer program product, embodied in a machine readable storage device, for enhancing a quality of data stored in a system, the computer program product comprising instructions for causing a processor to:

store data input in a data entry format via an interface, wherein the data in the data entry format excludes information required by a data entry rule;

transform the stored data from the data entry format to a first data storage format, wherein the data in the first data storage format includes the information required by the data entry rule;

identify an error in the data in the first data storage format, the error belonging to a particular type of error;

route the data to a selected one of first and second error correctors based on a type of the error;

receive corrected data from the selected one of the first and second error correctors; [[~~and~~]]

store the corrected data in the first database in the first data storage format;

receive data identifying a user who input the data via the interface;

derive additional data to be stored in the first data storage format based on the input data and based on an identity of the user; and

store the additional data in the first database.

10. (Previously Presented) The computer program product of claim 9, wherein transforming is performed by a rules-based procedure.

11. (Previously Presented) The computer program product of claim 9, wherein the computer program product further comprises instructions for causing a processor to provide default data values in the interface.

12-13. (Cancelled)

14. (Currently Amended) The computer program product of claim [[~~12~~]] 9, wherein the computer program product further comprises instructions for causing a processor to dynamically define the data entry format based on an identity of the user.

15. (Previously Presented) The computer program product of claim 14, wherein the computer program product further comprises instructions for causing a processor to provide default data values to the interface.

16. (Previously Presented) The computer program product of claim 9, wherein the computer program product further comprises instructions for causing a processor to:

transform the data from the data entry format to a second data storage format; and

store the data in a second database in the second data storage format.

17. (Previously Presented) The method of claim 1, further comprising:

monitoring a workload of the first and second error correctors; and

shifting error handling responsibilities from the first error corrector to a different error corrector in response to detecting that the workload of the first error corrector is higher than a desired workload.

18. (Previously Presented) The method of claim 1, further comprising:

determining a desired timeframe for resolving the error; and

sending a reminder to the selected one of the first and second error correctors, the reminder including a request to resolve the error by the desired timeframe.

19. (Previously Presented) The computer program product of claim 9, further comprising instructions to:

monitor a workload of the first and second error correctors; and

shift error handling responsibilities from the first error corrector to a different error corrector in response to detecting that the workload of the first error corrector is higher than a desired workload.

20. (Previously Presented) The computer program product of claim 9, further comprising instructions to

determine a desired timeframe for resolving the error; and

send a reminder to the selected one of the first and second error correctors, the reminder including a request to resolve the error by the desired timeframe.